Unified Knowledge Distillation Framework: Fine-Grained Alignment and Geometric Relationship Preservation for Deep Face Recognition

Durgesh Mishra[†, ✉], Rishabh Uikey
Indian Institute of Science Education and Research, Bhopal, India
{durgesh.mishraa10, rishabhuikey}@gmail.com

Abstract

Knowledge Distillation (KD) is crucial for optimizing face recognition models for deployment in computationally limited settings, such as edge devices. Traditional KD methods, such as Raw L2 Feature Distillation or Feature Consistency (FC) loss, often fail to capture both fine-grained instance-level details and complex relational structures, leading to suboptimal performance. We propose a unified approach that integrates two novel loss functions: Instance-Level Embedding Distillation (ILED) and Relation-Based Pairwise Similarity Distillation (RPSD). ILED focuses on aligning individual feature embeddings by leveraging a dynamic hard mining strategy, thereby enhancing learning from challenging examples. RPSD captures relational information through pairwise similarity relationships, employing a memory bank mechanism and a sample mining strategy. This unified framework ensures both effective instance-level alignment and preservation of geometric relationships between samples, leading to a more comprehensive distillation process. Our unified framework outperforms state-of-the-art distillation methods across multiple benchmark face recognition datasets, as demonstrated by extensive experimental evaluations. Interestingly, when using strong teacher networks compared to the student, our unified KD enables the student to even surpass the teacher's accuracy.

1. Introduction

Face Recognition (FR) is integral to various security and authentication systems due to its quick and trustworthy identification performance. Earlier FR methods relied on traditional handcrafted features [44, 1, 24], which typically lacked generalization. The advent of deep convolutional neural networks (CNNs) [43, 33] significantly boosted FR performance, replacing traditional feature extraction approaches and achieving SoTA results on benchmark datasets [50, 29].

Furthermore, there is a growing trend towards deploying FR systems on low-computation edge devices [2], such as smartphones, wearables, and IoT devices, driven by the increasing demand for privacy, real-time processing, and cost-efficiency. However, this introduces significant challenges, as large architectures such as ResNet100 [16], Inception-ResNet-v2 [42], and SENet-154 [19], which achieve high accuracy, are incompatible with the limited computational resources and the restricted memory capabilities inherent in these devices.

To address these challenges, model compression techniques, such as knowledge distillation (KD) [18], pruning [22], and quantization [10], play a crucial role in optimizing FR systems for deployment on low-computation devices. This paper focuses specifically on the KD approach, in which a complex teacher model transfers its learned knowledge to a comparatively simpler student model. This knowledge transfer between the student and teacher models can occur in three main ways [11]: response-based, feature-based, and relation-based KD.

Traditional KD methods [18, 54] often rely on KL Divergence with Soft Logits as the primary distillation loss. Recent studies have shown that using L2 loss on raw unnormalized feature embeddings, also called Raw L2 Feature Distillation [23, 37, 28], and L2 loss on normalized feature embeddings, known as Feature Consistency (FC) loss [6, 25, 46], can significantly outperform KL-divergence-based methods for aligning feature embeddings. The FC loss is a simple, straightforward way to align the student model representation with that of the teacher, but this direct approach may not effectively capture fine-grained information, leading to suboptimal performance. Moreover, recent research in metric learning [14, 40, 52] emphasizes the importance of hard negative samples for enhancing the discriminative power of feature embeddings. Thus, a more refined approach that dynamically targets hard samples is needed for more effective knowledge transfer.

Additionally, while FC loss facilitates feature alignment

[†] Part of this work was done as a Research Fellow at DRDO Young Scientist Laboratory – AI. [✉] Corresponding author.

at the instance level, it fails to capture the relational information between different samples or the structural relationships learned by the teacher model. Existing methods [32, 4] have attempted to address this limitation by incorporating relational knowledge into the distillation process, but these methods lack dynamic hard mining strategies [8, 4] or suffer from static mining processes with significant computational overhead [25]. Given these limitations, there is a need for more advanced KD methods that can effectively capture both fine-grained instance-level details and complex relational structures, ensuring robust performance even on resource-constrained edge devices. Our proposed approach introduces two novel loss functions: Instance-Level Embedding Distillation (ILED) and Relation-Based Pairwise Similarity Distillation (RPSD); designed to overcome the shortcomings of existing techniques. The primary contributions of the paper are as follows:

- We propose a novel ILED loss function based on the rescaled softplus function, incorporating a dynamic sample mining strategy to effectively handle both easy and hard samples, ensuring better alignment between the student and teacher models.
- We introduce the RPSD loss function, which leverages pairwise similarity relationships to capture complex geometric relationships within the embedding space, enhancing the knowledge transfer by focusing on relational information between samples.
- Our approach integrates both ILED and RPSD losses into a unified framework, balancing fine-grained instance-level details with broader relational structures, leading to improved performance in knowledge distillation tasks.

The remainder of the paper is structured as follows: Section 2 provides an overview of related work on FR losses and distillation methods for FR. Section 3 describes the details of the proposed method. Section 4 outlines the experimental setup and presents the results, comparing the proposed approach with existing methods. Finally, Section 5 concludes the paper.

2. Related Work

This section reviews FR literature and the KD methods applied to it.

2.1. Face Recognition Losses

In FR, two widely used categories of loss functions are metric learning and angular softmax-based losses [47]. Some common losses used in metric learning are Contrastive Loss [13] and Triplet Loss [38, 17]. In addition to metric learning losses, FR models are also trained using softmax-based loss functions, which are treated as a classification problem. However, previous studies [43, 41] have shown that features learned through softmax loss are merely separable, lacking sufficient discriminative power. To overcome this limitation, L-Softmax [27] was introduced aiming to push the decision boundaries between classes further apart in the loss function.

Further, SphereFace [26] refines this by normalizing the weights and incorporating a more rigorous angular margin, creating a spherical decision boundary. CosFace [45] simplifies this approach by using a cosine margin, directly optimizing the cosine similarity between features and class centers, which leads to better intra-class compactness. ArcFace [5] extends these ideas further by applying an additive angular margin in the arccosine space, ensuring that features lie on a normalized hypersphere, which enhances both inter-class separability and intra-class compactness.

MagFace [30] and AdaFace [21] both use feature norms as a proxy for image quality: higher norms mean high quality samples and lower norms mean low quality or hard samples. MagFace extends ArcFace with a feature norm-aware adaptive margin and a regularization term, encouraging high quality samples to have larger norms (closer to class centers) and pushing low quality ones away (smaller norms). AdaFace, by contrast, treats norms as fixed quality indicators for dynamic loss weighting, giving more weight to moderately hard but recognizable faces and less weight to unrecognizable ones without explicitly enforcing norm changes during backpropagation. SphereFace2 [49] emphasizes four key principles for binary classification-based training, transforming a k-class classification problem into k-binary classification tasks. A key component in the aforementioned methods is the normalization of features and weight vectors, which ensures that the learned features are distributed on a hypersphere, contributing to the robustness and effectiveness of the model.

2.2. Knowledge Distillation in Face Recognition

We now discuss the KD methods used for training the student model. DarkRank [4] trains the student to mimic the teacher's relative geometry within each mini-batch. For a query (the first sample in a mini-batch of size m), it measures Euclidean distances to the other $m-1$ samples, converts them into scores, and sorts them to form the teacher's ranking. The student does the same, and the loss forces its ranking to match the teacher's: the soft version aligns the full distribution over all orderings, while the hard version only aligns the top-ranked list.

Grouped Knowledge Distillation [54] splits teacher logits into Primary (most informative) and Secondary (less informative) groups, then uses a Binary-KD loss on the Primary Group to align student distributions. Attention Similarity Knowledge Distillation (A-SKD) [39] transfers atten-

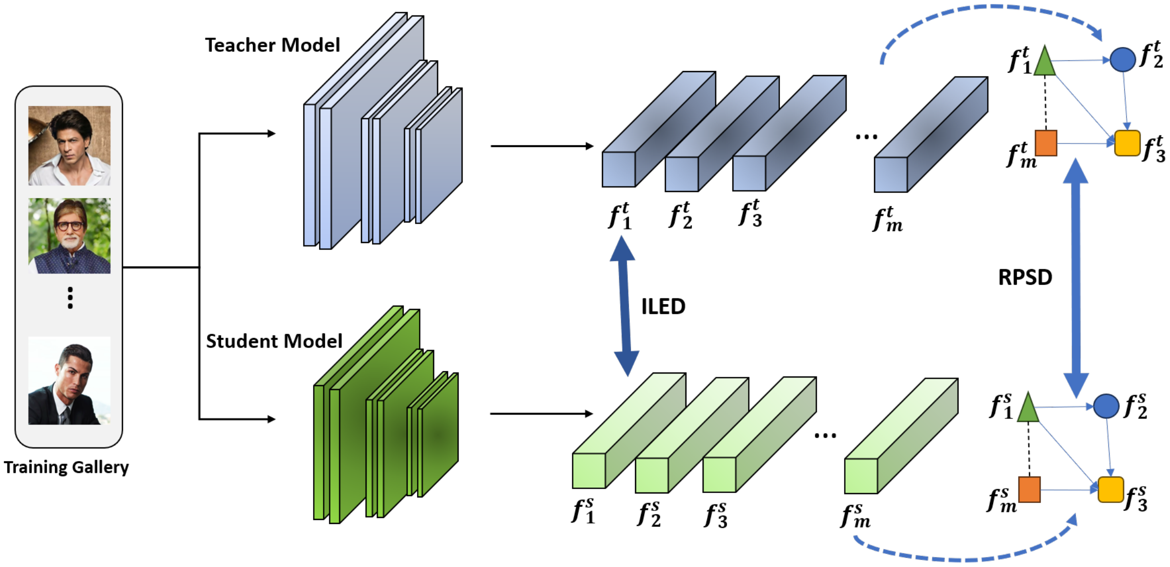

Figure 1. System overview of the unified knowledge distillation framework: Instance-Level Embedding Distillation (ILED) dynamically aligns hard samples, while Relation-Based Pairwise Similarity Distillation (RPSD) preserves geometric relationships through hard-mining and a memory bank, jointly optimizing face recognition models.

tion maps from a high-resolution teacher network to a low-resolution student network using a Convolutional Block Attention Module (CBAM) [51]. Relational Knowledge Distillation (RKD) focuses on transferring the relational information between pairs or groups of data examples rather than individual outputs [32]. This approach introduces two methods: Distance-wise Distillation and Angle-wise Distillation.

Correlation Congruence for Knowledge Distillation (CCKD) [35] improves student network performance by transferring instance-level information and correlations from a teacher network. Exclusivity-Consistency Regularized Knowledge Distillation (EC-KD) [46] incorporates Weight Exclusivity Regularization (diversifying convolutional filters) and Feature Consistency Regularization (aligning intermediate features instead of logits) for FR. ShrinkTeaNet [6] introduces Angular Distillation loss, where the student matches only the direction (not the magnitude) of each feature embedding with that of the teacher on a hypersphere. The KD loss is applied hierarchically at the final and intermediate layers with exponentially decaying weights.

CoupleFace [25] combines Feature Consistency Distillation (FCD) and Mutual Relation Distillation (MRD) into a distillation loss. FCD uses the L2 distance on normalized feature embeddings to measure the difference between the teacher and student models. MRD captures mutual relation knowledge by evaluating the relative distances (cosine similarities) between pairs of samples. Hybrid Order Relational Knowledge Distillation (HORKD) [8] introduces a teacher stream trained on high-resolution images, a student stream learning to mimic the teacher on low-resolution images, and an assistant stream facilitating knowledge transfer. The method uses a loss function incorporating multiple levels of relational knowledge (1-order, 2-order, 3-order, and center-based). The approach in [25] assumes that FCD loss will rapidly align the student and teacher embeddings, but this convergence may be imperfect in complex datasets or limited training conditions. If convergence is inadequate, the mutual relation $R(f_i^s, f_j^t)$ might not correctly approximate $R(f_i^s, f_j^s)$, resulting in suboptimal knowledge transfer and performance. While the additional “assistant” stream in [8] improves knowledge transfer, it also introduces more complexity into the training process.

AdaDistill [7] guides learning by matching each student embedding to an adaptive positive prototype: initially, the prototype closely resembles the teacher’s feature, encouraging sample-level imitation, but an exponential moving average gradually shifts it toward the teacher’s class centroid. Later training uses only the averaged prototype, ignoring intra-class variations (pose, age) and sample-to-sample relationships. Since the student trains solely with this centre-based margin loss [5, 45], without auxiliary cross-entropy, an inaccurate or shifting class centre may negatively impact learning, particularly for multi-modal classes or noisy data.

3. Proposed Method

This section provides an overview of our unified KD framework in FR, as illustrated in Fig. 1. Our approach combines Instance-Level Embedding Distillation and Relation-Based Pairwise Similarity Distillation to enhance the transfer of knowledge from a teacher model to a student model. Together, these methods aim to ensure more comprehensive and robust knowledge distillation.

3.1. Instance-Level Embedding Distillation (ILED)

ILED focuses on individual examples within the training dataset. This involves training the student model to align its feature embeddings with those generated by the teacher model for each face image. For notations, let $\mathbf{f}^t \in \mathbb{R}^d$ and $\mathbf{f}^s \in \mathbb{R}^d$ denote the feature embeddings of the teacher and student models, where d represents the dimensionality of the embeddings. Traditional FR methods employ a FC loss [6, 25], shown in (1). The FC loss minimizes the L2 norm of the difference between the normalized teacher and student embeddings, enabling the student model to closely align its features with those of the teacher model.

Consider the L2 norm of the difference between the normalized teacher and student embeddings for a batch of size m.

$$\mathcal{L}_{\text{FC}} = \frac{1}{m}\sum_{i=1}^{m}\left\|\frac{\mathbf{f}_i^t}{\|\mathbf{f}_i^t\|} - \frac{\mathbf{f}_i^s}{\|\mathbf{f}_i^s\|}\right\|^2 \tag{1}$$

Upon simplification, this expression can be reformulated as

$$\mathcal{L}_{\text{FC}} = \frac{2}{m}\sum_{i=1}^{m}(1 - x_i). \tag{2}$$

Where x_i, representing the cosine similarity between the teacher and student embeddings for image i, is given by

$$x_i = \frac{\mathbf{f}_i^t \cdot \mathbf{f}_i^s}{\|\mathbf{f}_i^t\|\,\|\mathbf{f}_i^s\|}. \tag{3}$$

Equation (2) provides a straightforward measure for aligning the student embeddings with the teacher embeddings. However, such a direct approach may fail to capture the fine-grained information effectively, particularly when training a comparatively simpler student model. The simplistic use of cosine similarity as the loss function can overlook subtle nuances and intricate patterns present in the data, leading to suboptimal knowledge transfer. Several recent studies in metric learning [14, 40, 52, 21, 30] have shown that hard negative samples are important for improving the discriminative power of feature embeddings. Therefore, a more refined approach is needed, one that incorporates additional strategies to dynamically focus on the most challenging examples, thereby ensuring a more comprehensive and accurate distillation of knowledge.

Dynamic Sample Mining Strategy in ILED. To implement this refined approach, we employ a dynamic sample mining strategy that adjusts the contribution of each sample to the loss function based on its difficulty. Cosine similarity ranges from -1 to 1, with 1 indicating perfect alignment between the embeddings. Samples with high cosine similarity (close to 1) are considered *"easy"* because they indicate that the student model's embeddings are already well-aligned with the teacher's embeddings. These easy samples result in a lower contribution to the loss, allowing the model to focus more on challenging examples. Conversely, samples with lower cosine similarity (further from 1) are considered *"hard"* as they represent instances where the student model struggles to accurately replicate the teacher's embeddings. These hard samples contribute more significantly to the loss, directing the model's attention toward areas that require improvement.

To achieve this objective, we propose an ILED loss function that builds upon the rescaled softplus function [9], incorporating a dynamic easy/hard sample mining strategy, defined as:

$$\mathcal{L}_{\text{ILED}} = \frac{1}{r}\ln\big(1 + \exp(-r(\bar{x} - s))\big)\sqrt{(\bar{x} - s)^2 + b}\,. \tag{4}$$

Where $\bar{x}$ represents the average cosine similarity ($\cos\theta$) between the student and teacher feature embeddings in a batch, given as

$$\bar{x} = \frac{1}{m}\sum_{i=1}^{m} x_i\,. \tag{5}$$

The hyperparameter s serves as a soft margin, providing a flexible target similarity that applies continuous penalties for deviations. Unlike a hard margin with a strict cutoff, s smoothly guides the model toward the desired similarity. The hyperparameter r controls the steepness of the loss curve around the soft margin term s. Higher values of r make the function more sensitive to deviations from s, emphasizing hard samples. A small constant b is used to ensure numerical stability and maintain smoothness. To control the relative contributions of ILED loss, a hyperparameter λ_{ILED} is used. More importantly, ILED matches student and teacher embeddings solely through cosine similarity, so both vectors are L2-normalised onto the unit hypersphere; alignment depends only on direction and ignores norms, granting the student extra degrees of freedom to interpret its teacher's knowledge during learning.

Fig. 2 shows a comparison between the traditional FC loss and the proposed ILED loss for varying scaling factors, λ_{ILED}. The ILED loss uses hyperparameters such as r (steepness), s (soft margin), and b (smoothness) to dynamically adjust the loss based on sample difficulty. This approach emphasizes harder samples—those with lower cosine similarity, by assigning them a higher loss value, while easier samples—those with higher cosine similarity, contribute less to the overall loss. Overall, there are two main

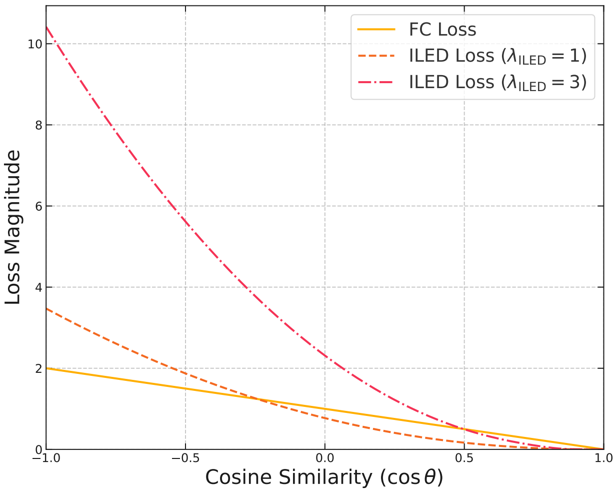

Figure 2. The plot illustrates the FC loss and ILED losses at different scaling factors λ_{ILED} while keeping the hyper-parameters fixed at $s = 0.85$, $b = 0.1$, and $r = 40$.

terms in (4), a *"logarithmic term"* and a *"distance weighting component"*. The logarithmic term measures the misalignment between the average cosine similarity score $\bar{x}$ and a desired target value s, smoothly penalizing deviations while allowing for gradient-based optimization. This term encourages the model to align its outputs closely with the target similarity. The distance weighting component dynamically adjusts the importance of each sample based on its distance from the target s, amplifying the contribution of hard samples (those further from the target) and minimizing the impact of easy samples (those closer to the target). Together, these terms ensure a balanced focus, guiding the model to learn effectively by correcting difficult cases while not over-penalizing already well-aligned examples.

3.2. Relation-Based Pairwise Similarity Distillation (RPSD)

While ILED focuses on individual embeddings, RPSD takes a different approach by leveraging the relational structure between feature embeddings from a teacher model to a student model using the pairwise relationships between samples. This approach enhances knowledge transfer by matching the pairwise cosine similarities of teacher and student feature embeddings, ensuring the student learns the underlying geometric structure of the teacher's embedding space.

Pairwise Relations and Higher-Order Dependencies. In angular margin-based loss functions, embeddings are normalized and projected onto a hypersphere while training [45, 5]. Since the feature embeddings are constrained to lie on the surface of a unit hypersphere, the angles (or cosine similarities) between them are sufficient to describe their geometric relationships. Thus, it is reasonable to assume that matching the cosine similarity relations between samples from the student and teacher models will effectively capture the underlying knowledge.

For normalized embeddings, higher-order relations (such as those between triplets) can be seen as combinations of pairwise relations, which simplifies the distillation process by focusing on matching these pairwise relations between the student and teacher models. To illustrate this, consider the example of a triplet relation involving three normalized vectors $\mathbf{f}_t^i, \mathbf{f}_t^j, \mathbf{f}_t^k \in \mathbb{R}^d$, which represent feature embeddings from the teacher model. The angle between any two of these vectors can be expressed using pairwise cosine similarities:

$$\cos\theta_{ij} = \mathbf{f}_t^i \cdot \mathbf{f}_t^j, \quad \cos\theta_{jk} = \mathbf{f}_t^j \cdot \mathbf{f}_t^k, \quad \cos\theta_{ik} = \mathbf{f}_t^i \cdot \mathbf{f}_t^k, \tag{6}$$

where θ_{ij}, θ_{jk}, and θ_{ik} are the angles between the respective vectors.

To compute a higher-order relation, such as the angle formed by the three vectors, we consider the angle between the vectors $\mathbf{f}_t^i - \mathbf{f}_t^j$ and $\mathbf{f}_t^k - \mathbf{f}_t^j$. This can be expressed as:

$$\cos\angle\mathbf{f}_t^i\mathbf{f}_t^j\mathbf{f}_t^k = \frac{(\mathbf{f}_t^i - \mathbf{f}_t^j)\cdot(\mathbf{f}_t^k - \mathbf{f}_t^j)}{\|\mathbf{f}_t^i - \mathbf{f}_t^j\|\,\|\mathbf{f}_t^k - \mathbf{f}_t^j\|}. \tag{7}$$

By expanding the dot product and utilizing the normalization of the vectors, we arrive at the following

$$(\mathbf{f}_t^i - \mathbf{f}_t^j)\cdot(\mathbf{f}_t^k - \mathbf{f}_t^j) = \cos\theta_{ik} - \cos\theta_{ij} - \cos\theta_{jk} + 1\,. \tag{8}$$

The norms of the differences can be computed as

$$\begin{aligned} \|\mathbf{f}_t^i - \mathbf{f}_t^j\| &= \sqrt{2(1-\cos\theta_{ij})}, \\ \|\mathbf{f}_t^j - \mathbf{f}_t^k\| &= \sqrt{2(1-\cos\theta_{jk})}. \end{aligned} \tag{9}$$

Substituting expression (8) and (9) into (7) yields the final form:

$$\cos\angle\mathbf{f}_t^i\mathbf{f}_t^j\mathbf{f}_t^k = \frac{\cos\theta_{ik} - \cos\theta_{ij} - \cos\theta_{jk} + 1}{\sqrt{2(1-\cos\theta_{ij})}\,\sqrt{2(1-\cos\theta_{jk})}}. \tag{10}$$

This example demonstrates that the angle formed by three normalized vectors, such as feature embeddings, can be entirely expressed using pairwise cosine similarities $\cos\theta_{ij}$, $\cos\theta_{jk}$, and $\cos\theta_{ik}$. Therefore, matching the pairwise relationships between the student and teacher models should provide sufficient information for effective knowledge distillation. Moreover, since the vectors are normalized, calculating the pairwise similarity is simply done by computing the dot product between the matrices. This approach simplifies the calculations and is computationally efficient.

Memory Bank Mechanism for RPSD. In transferring relational structure information from a teacher model to a student model, it is essential to capture the relationships between all possible pairs of samples in a dataset. However, computing all possible pairs in a dataset to capture relational information is computationally infeasible, especially in large datasets, due to the quadratic growth in the number of pairs as the dataset size increases. Consequently, the size of mini-batches imposes a practical limit on the extent of pairwise coverage that can be achieved. This constraint can lead to suboptimal learning of the student model, as the similarity information is only derived from a small set of examples at a time. To address this limitation, we employ a memory bank mechanism that maintains dynamic storage of feature embeddings using a queue and dequeue strategy [15]. Specifically, we use a first-in-first-out (FIFO) queue where the oldest mini-batch is dequeued when the current mini-batch is enqueued. This mechanism allows the computation of similarities between the current mini-batch and a more extensive pool of data available in the memory bank.

Let the size of each mini-batch be m, and the total capacity of the memory bank be q. The embeddings of the teacher and student models stored in the memory banks are represented as $\mathbf{F}_t = [\mathbf{f}_t^1, \mathbf{f}_t^2, \dots, \mathbf{f}_t^q] \in \mathbb{R}^{q \times d}$ and $\mathbf{F}_s = [\mathbf{f}_s^1, \mathbf{f}_s^2, \dots, \mathbf{f}_s^q] \in \mathbb{R}^{q \times d}$, respectively, where d is the embedding dimension.

For each training iteration, we compute the cosine similarities between the embeddings of the current mini-batch and those stored in the memory bank for student and teacher models. Let the embeddings of the current mini-batch be denoted by $\mathbf{E}_t = [\mathbf{e}_t^1, \mathbf{e}_t^2, \dots, \mathbf{e}_t^m] \in \mathbb{R}^{m \times d}$ for the teacher model and $\mathbf{E}_s = [\mathbf{e}_s^1, \mathbf{e}_s^2, \dots, \mathbf{e}_s^m] \in \mathbb{R}^{m \times d}$ for the student model. The cosine similarity matrices are defined as:

$$\mathbf{S}_t(i,j) = \frac{\mathbf{e}_t^i \cdot (\mathbf{f}_t^j)^\top}{\|\mathbf{e}_t^i\| \|\mathbf{f}_t^j\|}, \quad \mathbf{S}_s(i,j) = \frac{\mathbf{e}_s^i \cdot (\mathbf{f}_s^j)^\top}{\|\mathbf{e}_s^i\| \|\mathbf{f}_s^j\|}, \tag{11}$$

where $\mathbf{S}_t \in \mathbb{R}^{m \times q}$ and $\mathbf{S}_s \in \mathbb{R}^{m \times q}$ are the cosine similarity matrices of the teacher and student models, respectively. These matrices capture the pairwise cosine similarities between the embeddings in the current mini-batch and all the embeddings stored in the memory bank.

To quantify the difference in the relational feature spaces of the teacher and student models, the absolute element-wise difference between the corresponding similarity matrices is calculated.

$$\mathbf{D}(i,j) = |\mathbf{S}_t(i,j) - \mathbf{S}_s(i,j)| \tag{12}$$

The resulting matrix $\mathbf{D} \in \mathbb{R}^{m \times q}$ quantifies the dissimilarity between the teacher and student models based on their pairwise cosine similarities, capturing the alignment of their feature representations. The elements of $\mathbf{D}$ range from 0 to 2 but tend to be close to 0, even when the vectors are dissimilar, due to the properties of the cosine similarity difference. We use absolute differences instead of squared differences, as squaring smaller values would result in even smaller values, thus reducing the sensitivity of the measure to subtle variations between the models' embeddings.

The total dissimilarity across all pairs of embeddings is obtained by summing all elements of the matrix $\mathbf{D}$.

$$\text{Total Dissimilarity} = \sum_{i=1}^{m} \sum_{j=1}^{q} \mathbf{D}(i,j) \tag{13}$$

To make the loss scale invariant and comparable across different batch sizes and memory bank capacities, the total dissimilarity is normalized by the number of elements ($m \times q$) in the matrix, resulting in the Normalized Dissimilarity, denoted as Δ_{norm}. This normalization produces a single scalar value that represents the average discrepancy between the similarity matrices of the student and teacher models, facilitating consistent and fair comparisons.

Dynamic Sample Mining Strategy in RPSD. To effectively handle both easy and hard samples, we adopt a dynamic sample mining strategy, same as (4), that adjusts the contribution of each sample to the loss based on its normalized dissimilarity value. A sample with Δ_{norm} very close to 0 is considered *"easy,"* indicating that the student model has effectively captured the teacher's relational structure and thus contributes less to the overall loss, allowing the model to prioritize more hard samples. Conversely, samples with Δ_{norm} values that deviate further from 0 are considered *"hard"* samples, representing instances where the student model struggles to replicate the relational structure of the teacher. These hard samples indicate areas where the student model has not yet learned effectively, and therefore, they contribute more significantly to the loss. This strategy ensures that the student model focuses on learning from the most challenging examples, driving improvement where it is most needed.

To dynamically adjust the impact of each sample based on its normalized similarity difference, we propose the RPSD loss function, capturing the relational structure between samples:

$$\mathcal{L}_{\text{RPSD}} = \frac{1}{r'} \log\left(1 + \exp\left(r' \cdot (\Delta_{\text{norm}} - t)\right)\right) \cdot \sqrt{(\Delta_{\text{norm}} - t)^2 + b'} \tag{14}$$

Where Δ_{norm} represents the Normalized Dissimilarity across all samples, and r', t, and b' are hyperparameters that control the shape and scale of the loss function. Here, t serves as a threshold or transition parameter that defines when the loss begins to increase significantly, distinguishing between *"easy"* and *"hard"* samples. The logarithmic term helps control or dampen the contribution of *"easy"*

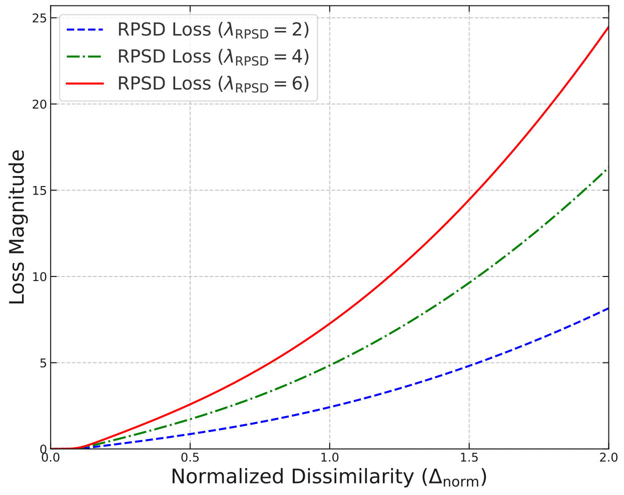

Figure 3. The plot illustrates the RPSD loss at different scaling factors λ_{RPSD} while the hyper-parameters are fixed to $t = 0.1$, $b' = 1$, and $r' = 60$.

samples, ensuring that the loss does not overly penalize the model for minor deviations. In contrast, the square root term enhances the emphasis on *"hard"* samples, directing the model's learning focus toward areas where the alignment between the teacher and student embeddings is weaker. Together, these terms balance the loss function's response to both easy and hard samples. A hyperparameter λ_{RPSD} controls the relative contributions of the RPSD loss.

Fig. 3 illustrates the behavior of the RPSD loss for different scaling factors, λ_{RPSD}, across the range of normalized dissimilarity values from 0 to 2. The plot demonstrates how the RPSD loss function emphasizes hard samples (higher dissimilarity values) by increasing the loss magnitude while still allowing for dynamic adjustment based on the chosen scaling factor. This formulation ensures that the student model aligns its internal relational structure with that of the teacher model.

3.3. Unified Loss Function

The unified KD loss function for training the student model is defined as a weighted combination of the ILED loss, RPSD loss, and standard FR loss:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{ILED}} \cdot \mathcal{L}_{\text{ILED}} + \lambda_{\text{RPSD}} \cdot \mathcal{L}_{\text{RPSD}} + \mathcal{L}_{\text{FR}} \qquad (15)$$

Where $\mathcal{L}_{\text{ILED}}$ is the Instance-Level Knowledge Distillation loss as defined in (4), $\mathcal{L}_{\text{RPSD}}$ is the Relation-Based Knowledge Distillation loss as defined in (14), and $\mathcal{L}_{\text{FR}}$ is the standard FR loss, which helps the student model learn the correct class labels directly from the training data. The parameters λ_{ILED} and λ_{RPSD} are hyperparameters that control the relative contributions of each loss component.

Refer to Appendix A for a detailed, step-by-step pseudocode of our unified KD framework (ILED + RPSD). It details every stage of training, including KD loss calculations, memory-bank updates, and parameter updates.

4. Experiments and Results

Datasets. For training, a preprocessed VGGFace2 dataset [3] is used in all methods. In our experiments with MS1M-V2, a refined version of the MS-Celeb-1M dataset [12, 5] and VGGFace2, we observed that VGGFace2 converged faster and achieved a lower training loss in fewer iterations. This is likely due to its cleaner labels and greater intra-class variability, while MS1M-V2 requires more iterations due to its larger dataset size and higher noisy labels. The preprocessed VGGFace2 dataset contains 3.1 million images from 8.6 thousand identities. The preprocessing process for the VGGFace2 dataset [49] involves cropping each face image using a similarity transformation based on five facial landmarks detected by MTCNN [53]. This results in images of size 112 × 112 pixels. Additionally, each RGB pixel value, originally ranging from [0, 255], is normalized to the range of $[-1, 1]$. For validation, four datasets are used: LFW [20], AgeDB [31], CA-LFW [56], and CP-LFW [55], each containing 3,000 positive pairs and 3,000 negative pairs. For testing, two large-scale IARPA Janus benchmarks are used: IJB-B[50] (21,798 images and 55,026 frames from 7,011 videos, 1,845 identities) and its extension IJB-C [29] (31,334 images and 117,542 frames from 11,779 videos, 3,531 identities).

Experimental setting. All methods are implemented using the OpenSphere GitHub repository [48] and using the PyTorch framework [34]. For a fair comparison, ResNet100 is used as the teacher's backbone, and ResNet18 is used as the student's backbone [16]. SphereFace2 [49] is employed as the traditional FR loss to train both the student and the teacher models. In experimentation on the VGGFace2 dataset, SphereFace2 showed slightly better performance compared to other state-of-the-art loss functions, including CosFace [45], ArcFace[5], SphereFace[26], and AdaFace[21]. The hyperparameters used for training are consistent with those reported in the paper: λ = 0.7, r=40, m=0.4, t=3.0, where m is the CosFace-type additive margin. We use Stochastic Gradient Descent (SGD) [36] with a momentum of 0.9 and a batch size of 64. A Step Decay learning rate schedule is applied, starting at 0.1 and reducing by a factor of 10 at 50K, 100K, 120K, and 140K iterations. For testing, the model weights from the 140K iteration checkpoint are used. The experiments were conducted using NVIDIA RTX 4090 and RTX 6000 Ada GPUs to perform all computations.

Table 1. Consolidated performance comparison of knowledge-distillation methods. Left: validation accuracy (%) on LFW, AgeDB, CA-LFW, and CP-LFW (higher is better). Right: verification rate (VR %) on IJB-B and IJB-C at FAR = 10^{-5} and 10^{-4} (higher is better). *Student model trained without any distillation, using only the SphereFace2 loss.

Method	Validation accuracy (%)				IJB-B VR (%)		IJB-C VR (%)	
	LFW	AgeDB	CA-LFW	CP-LFW	10^{-5}	10^{-4}	10^{-5}	10^{-4}
Student model*	99.417	93.367	93.417	90.917	79.903	89.085	85.749	91.266
Teacher model	99.683	95.633	94.533	93.117	85.910	92.142	90.377	94.089
KL-DIV [18]	99.450	92.517	93.317	89.800	77.945	87.215	83.438	89.692
Raw L2 [23, 37]	99.383	93.167	93.533	89.783	77.556	87.546	83.876	89.876
FC loss [25, 46]	99.533	94.483	93.850	91.300	81.782	89.552	86.516	91.686
DarkRank [4]	99.550	94.167	93.683	90.883	79.659	88.695	85.514	91.190
ShrinkTeaNet [6]	99.533	94.000	93.767	91.333	82.269	89.942	87.084	92.182
CCKD [35]	99.567	94.417	93.950	91.167	82.123	89.942	87.202	92.120
EC-KD [46]	99.583	93.283	93.583	90.800	79.474	89.114	85.478	91.251
ILED only	99.583	94.617	**94.033**	91.800	82.317	89.912	86.823	91.906
RPSD only	99.567	94.583	93.850	90.783	80.282	89.279	86.036	91.543
Unified KD	**99.617**	**94.817**	94.000	**91.900**	**82.405**	**90.117**	**87.288**	**92.458**

4.1. Results on the Validation and Test Datasets

In this section, we validate the effectiveness of our proposed methods—ILED only, RPSD only, and Unified KD framework, by comparing them against several baseline methods commonly used in KD methods for FR tasks. We compare our method with KL Divergence with Soft Logits [18], Raw L2 Feature Distillation (Raw L2) [23, 37], FC loss [6, 25, 46], DarkRank [4], ShrinkTeaNet [6], Grouped Knowledge Distillation (GKD) [54], Correlation Congruence KD (CCKD) [35], Exclusivity-Consistency Regularized KD (EC-KD)[46], and AdaDistill [7]. Refer to Appendix C.3 for the full hyperparameter settings of all baseline KD methods.

For our proposed methods, the parameters for the ILED method were set as follows: $r = 40$, $s = 0.9$, $b = 0.1$, and λ_{ILED} = 3. For the RPSD method, the parameters were configured with $r' = 60$, $t = 0.05$, $b = 1$, and λ_{RPSD} = 40. The memory bank size q is set to be three times the batch size, and the feature embedding dimension for both the student and teacher models is 512. The results on the validation datasets (LFW, AgeDB, CA-LFW, and CP-LFW) and on the test datasets (IJB-B and IJB-C) for the aforementioned methods are all shown in Table 1. The numbers for the GKD and AdaDistill methods were very low, so we did not include them in the comparison table. AdaDistill is specifically designed for margin-penalty Softmax losses (e.g., ArcFace and CosFace), making it incompatible with SphereFace2's binary angular-margin loss. Directly applying AdaDistill to SphereFace2 yields suboptimal results.

The results in Table 1 show that the unified approach of the ILED + RPSD methods consistently outperforms other knowledge-distillation techniques across multiple datasets. Specifically, it achieves the highest accuracy on LFW (99.617 %), AgeDB (94.817 %), and CP-LFW (91.900 %), closely matching the teacher-model performance, and it delivers the best verification rates at lower false-accept rates on both IJB-B and IJB-C, demonstrating its robustness and effectiveness compared with the student model without distillation and with other methods. Overall, this unified approach proves to be the most effective.

We conducted a small ablation study to evaluate the contributions of ILED and RPSD methods within our unified KD framework; see Appendix B. In Appendix C.1, we present results for different teacher–student pairings, illustrating how the capacity gap between models affects knowledge transfer. Appendix C.2 then examines the role of the teacher's pretraining loss. Together, these analyses demonstrate the robustness of our method across both architectural and loss-function variations.

5. Conclusion

In this paper, we propose a unified framework for KD in deep FR that combines two novel loss functions: ILED and RPSD. ILED dynamically focuses on challenging samples for better alignment, while RPSD captures relational structures to enhance the student's understanding of geometric relationships. Experiments on multiple benchmark datasets showed that our unified KD approach outperforms traditional FR methods, but the need to tune several hyperparameters is a limitation. Future work could focus on making the hyperparameters adaptive during training and the extension of our unified framework to other metric-learning tasks such as Person Re-Identification and Fine-Grained Image Retrieval.

References

[1] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[2] E. Caldeira, P. C. Neto, M. Huber, N. Damer, and A. F. Sequeira. Model compression techniques in biometrics applications: A survey, 2024.

[3] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, page 67–74. IEEE Press, 2018.

[4] Y. Chen, N. Wang, and Z. Zhang. Darkrank: accelerating deep metric learning via cross sample similarities transfer. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[5] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019.

[6] C. N. Duong, K. Luu, K. G. Quach, and N. T. H. Le. Shrinkteanet: Million-scale lightweight face recognition via shrinking teacher-student networks. *ArXiv*, abs/1905.10620, 2019.

[7] N. D. Fadi Boutros, Vitomir Štruc. Adadistill: Adaptive knowledge distillation for deep face recognition. In *Computer Vision - ECCV 2024 -18th European Conference on Computer Vision, Milano, Italy, September 29- 4 October, 2024*, October 2024.

[8] S. Ge, K. Zhang, H. Liu, Y. Hua, S. Zhao, X. Jin, and H. Wen. Look one and more: Distilling hybrid order relational knowledge for cross-resolution image recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):10845–10852, Apr. 2020.

[9] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In G. Gordon, D. Dunson, and M. Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 315–323, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR.

[10] Y. Gong, L. Liu, M. Yang, and L. Bourdev. Compressing deep convolutional networks using vector quantization, 2014.

[11] J. Gou, B. Yu, S. J. Maybank, and D. Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, Mar. 2021.

[12] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-Celeb-1M: A large-scale Celebrity face dataset. In *ECCV*, 2016.

[13] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[14] B. Harwood, V. K. B. G, G. Carneiro, I. Reid, and T. Drummond. Smart mining for deep metric learning, 2017.

[15] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning, 2020.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015.

[17] A. Hermans, L. Beyer, and B. Leibe. In defense of the triplet loss for person re-identification, 2017.

[18] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network, 2015.

[19] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, June 2018.

[20] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller. Labeled Faces in the Wild: A Database forStudying Face Recognition in Unconstrained Environments. In *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition*, Marseille, France, Oct. 2008. Erik Learned-Miller and Andras Ferencz and Frédéric Jurie.

[21] M. Kim, A. K. Jain, and X. Liu. Adaface: Quality adaptive margin for face recognition. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18729–18738, 2022.

[22] Y. LeCun, J. Denker, and S. Solla. Optimal brain damage. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann, 1989.

[23] J. Li, Z. Guo, H. Li, S. Han, J.-w. Baek, M. Yang, R. Yang, and S. Suh. Rethinking feature-based knowledge distillation for face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20156–20165, June 2023.

[24] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Transactions on Image Processing*, 11(4):467–476, 2002.

[25] J. Liu, H. Qin, Y. Wu, J. Guo, D. Liang, and K. Xu. Coupleface: Relation matters for face recognition distillation. In S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, editors, *Computer Vision – ECCV 2022*, pages 683–700, Cham, 2022. Springer Nature Switzerland.

[26] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition, 2018.

[27] W. Liu, Y. Wen, Z. Yu, and M. Yang. Large-margin softmax loss for convolutional neural networks, 2017.

[28] P. Luo, Z. Zhu, Z. Liu, X. Wang, and X. Tang. Face model compression by distilling knowledge from neurons. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), Mar. 2016.

[29] B. Maze, J. Adams, J. A. Duncan, N. Kalka, T. Miller, C. Otto, A. K. Jain, W. T. Niggel, J. Anderson, J. Cheney, and P. Grother. Iarpa janus benchmark - c: Face dataset and protocol. In *2018 International Conference on Biometrics (ICB)*, pages 158–165, 2018.

[30] Q. Meng, S. Zhao, Z. Huang, and F. Zhou. Magface: A universal representation for face recognition and quality assess-

ment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14225–14234, 2021.

[31] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou. Agedb: The first manually collected, in-the-wild age database. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1997–2005, July 2017.

[32] W. Park, D. Kim, Y. Lu, and M. Cho. Relational knowledge distillation. *CoRR*, abs/1904.05068, 2019.

[33] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *British Machine Vision Conference*, volume 1, page 6. BMVA Press, 2015.

[34] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019.

[35] B. Peng, X. Jin, D. Li, S. Zhou, Y. Wu, J. Liu, Z. Zhang, and Y. Liu. Correlation congruence for knowledge distillation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5006–5015, Oct 2019.

[36] B. T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

[37] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets, 2015.

[38] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2015.

[39] S. Shin, J. Lee, J. Lee, Y. Yu, and K. Lee. Teaching where to look: Attention similarity knowledge distillation for low resolution face recognition. In S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, editors, *Computer Vision – ECCV 2022*, pages 631–647, Cham, 2022. Springer Nature Switzerland.

[40] Y. Suh, B. Han, W. Kim, and K. M. Lee. Stochastic class-based hard example mining for deep metric learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7244–7252, June 2019.

[41] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1891–1898, June 2014.

[42] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 4278–4284. AAAI Press, 2017.

[43] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[44] M. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 01 1991.

[45] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu. Cosface: Large margin cosine loss for deep face recognition, 2018.

[46] X. Wang, T. Fu, S. Liao, S. Wang, Z. Lei, and T. Mei. Exclusivity-consistency regularized knowledge distillation for face recognition. In A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, editors, *Computer Vision – ECCV 2020*, pages 325–342, Cham, 2020. Springer International Publishing.

[47] X. Wang, J. Peng, S. Zhang, B. Chen, Y. Wang, and Y. Guo. A survey of face recognition, 2022.

[48] Y. Wen. Opensphere. `https://github.com/ydwen/opensphere`, n.d. Accessed: [Aug. 2024].

[49] Y. Wen, W. Liu, A. Weller, B. Raj, and R. Singh. Sphereface2: Binary classification is all you need for deep face recognition, 2022.

[50] C. Whitelam, E. Taborsky, A. Blanton, B. Maze, J. Adams, T. Miller, N. Kalka, A. K. Jain, J. A. Duncan, K. Allen, J. Cheney, and P. Grother. Iarpa janus benchmark-b face dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 592–600, 2017.

[51] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon. Cbam: Convolutional block attention module, 2018.

[52] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krähenbühl. Sampling matters in deep embedding learning, 2018.

[53] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.

[54] W. Zhao, X. Zhu, K. Guo, X.-Y. Zhang, and Z. Lei. Grouped knowledge distillation for deep face recognition. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023.

[55] T. Zheng and W. Deng. Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. Technical Report 18-01, Beijing University of Posts and Telecommunications, February 2018.

[56] T. Zheng, W. Deng, and J. Hu. Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments, 2017.

Appendix

A. Pseudo-Algorithm

We present the pseudo-algorithm (Algorithm 1) for our proposed Unified Knowledge Distillation (KD) framework for face recognition. The algorithm outlines the step-by-step process of our approach to optimize the student's model performance. In this pseudo-algorithm, we assume that the teacher model is already trained using a standard face recognition loss function (SphereFace2). In the notation used, r, s, and b are hyperparameters for the ILED, while r', t, and b' are for the RPSD. The parameters λ and λ' are weight factors that control the contributions of the ILED and RPSD losses to the total KD loss. The size of the memory banks is denoted by q, and the updating of a new batch in the memory bank is done using a queue and dequeue strategy, specifically employing a first-in-first-out (FIFO) method.

Algorithm 1 Unified Knowledge Distillation Framework for Deep Face Recognition

```
Input: Student model $S$, Teacher model $T$, Training data $\mathcal{D} = \{x_i, y_i\}_{i=1}^{N}$, Hyperparameters $(\lambda, r, b, s, \lambda', r', b', t, q)$
Output: KD-optimized student model $S$
Procedure:
Initialize parameters for $S$ and $T$
Load pretrained weights for $T$ and freeze its parameters
Set student and teacher memory banks: $F_s \leftarrow$ None, $F_t \leftarrow$ None
Initialize $L_{\text{RPSD}} \leftarrow 0$
for each iteration in the training process do
    Sample mini-batch $(\{x_i\}, \{y_i\}) \subset \mathcal{D}$, where $i = 1, \ldots, m$
    Compute student and teacher embeddings for mini-batch: $E_s \leftarrow S(\{x_i\}),\ E_t \leftarrow T(\{x_i\})$
    Compute recognition loss for student: $L_{\text{student}}$
    Calculate cosine similarities: $x_i \leftarrow \cos(f_s^i, f_t^i),\ \forall (f_s^i, f_t^i) \in (E_s, E_t)$
    Compute ILED Loss:
        $L_{\text{ILED}} = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{r} \ln\left(1 + \exp(-r \cdot (x_i - s))\right) \cdot \sqrt{(x_i - s)^2 + b}$
    if memory banks are sufficiently populated then
        Compute teacher and student similarity matrices: $S_t \leftarrow \cos(E_t, F_t^T),\ S_s \leftarrow \cos(E_s, F_s^T)$
        Calculate pairwise similarity difference: $D \leftarrow |S_t - S_s|$
        Normalize dissimilarity:
            $\Delta_{\text{norm}} = \frac{1}{q \cdot m} \sum_{i=1}^{q} \sum_{j=1}^{m} D(i, j)$
        Compute RPSD Loss:
            $L_{\text{RPSD}} = \frac{1}{r'} \log\left(1 + \exp(r' \cdot (\Delta_{\text{norm}} - t))\right) \cdot \sqrt{(\Delta_{\text{norm}} - t)^2 + b'}$
    end if
    Total Loss:
    $L_{\text{KD}} \leftarrow \lambda L_{\text{ILED}} + \lambda' L_{\text{RPSD}}$
    $L_{\text{total}} \leftarrow L_{\text{student}} + L_{\text{KD}}$
    Update Student Model:
    Backpropagate and update parameters: $\theta_s \leftarrow \theta_s - \eta \nabla_{\theta_s} L_{\text{total}}$
    Update memory banks: $F_s \leftarrow$ FIFO$(F_s, E_s),\ F_t \leftarrow$ FIFO$(F_t, E_t)$
end for
return optimized student model $S$
```

B. Ablation Study

We conducted a small ablation study to assess the impact of each component in our Unified Knowledge Distillation framework by isolating Instance-Level Embedding Distillation (ILED) and Relation-Based Pairwise Similarity Distillation (RPSD). This study aims to discern the contribution of each method to the overall performance of the student model across multiple benchmark face recognition datasets.

We assessed several variations of our approach on four validation datasets: LFW, AgeDB, CA-LFW, and CP-LFW. In accordance with the experimental setup outlined in the paper, a ResNet100 serves as the teacher backbone, while a ResNet18 is utilized as the student backbone. The SphereFace2 hyperparameters for training were kept consistent with those reported: $\lambda = 0.7$, $r = 40$, $m = 0.4$, $t = 3.0$, where m represents the CosFace-type additive margin.

B.1. Instance-Level Embedding Distillation (ILED)

The proposed ILED method uses three main hyperparameters: r (steepness), s (soft target), and λ_{ILED} (weight). The parameter b (small positive constant) is kept constant at 0.1 throughout the experiments, and λ_{ILED} is set to 3 to maintain a balanced contribution to the overall loss. The performance of the ILED method with different configurations of these hyperparameters is presented in Table 2.

Table 2. Performance of ILED with Different Hyperparameters on Validation Datasets

λ_{ILED}	r	s	LFW	AgeDB	CA-LFW	CP-LFW
3	40	0.5	99.467	93.933	93.850	91.000
3	40	0.6	99.550	94.000	93.850	91.017
3	40	0.7	99.517	93.783	93.667	91.000
3	40	0.8	**99.617**	94.417	93.733	91.317
3	40	0.9	99.583	**94.617**	**94.033**	**91.800**
3	20	0.9	99.517	94.333	93.883	91.283
3	40	0.9	**99.583**	**94.617**	**94.033**	**91.800**
3	60	0.9	99.517	94.300	93.833	91.183

From Table 2, it is observed that the optimal performance for ILED is achieved with $s = 0.9$ and $r = 40$, suggesting that a higher soft target s improves the alignment between the student and teacher embeddings.

B.2. Relation-Based Pairwise Similarity Distillation (RPSD)

The proposed RPSD function uses three main hyperparameters: r' (steepness), t (transition parameter), and λ_{RPSD} (weight). The parameter b (small positive constant) is kept constant at 1 throughout the experiments, and λ_{RPSD} is set to 60 to maintain a balanced contribution to the overall loss. The performance of the RPSD method with different configurations of these hyperparameters is presented in Table 3.

Table 3. Performance of RPSD with Different Hyperparameters on Validation Datasets

λ_{RPSD}	r'	t	LFW	AgeDB	CA-LFW	CP-LFW
60	60	0.05	**99.517**	**94.167**	**93.717**	**90.867**
60	60	0.10	99.433	93.567	93.433	90.600
60	60	0.15	99.417	93.100	93.400	90.750
60	60	0.20	99.450	93.133	93.417	90.717
60	40	0.05	99.533	94.150	93.633	**91.183**
60	60	0.05	99.517	**94.167**	**93.800**	90.867
60	80	0.05	**99.550**	93.733	93.717	91.000

C. Experimentations

In this section, we present a comprehensive evaluation of our Unified Knowledge Distillation (Unified KD) framework across a variety of settings. We first examine the impact of different teacher–student capacity gaps in Section C.1, then analyze the effect of the teacher's pretraining loss in Section C.2, and provide implementation details for the baseline methods in Section C.3.

C.1. Performance Across Teacher-Student Combinations

Table 4 compares students trained without any distillation against our Unified KD method—which combines ILED and RPSD, across three teacher architectures (ResNet100, ResNet50, DPN98) and four student backbones (ResNet18, MobileFaceNet, ResNet34, ResNet50). All networks were trained using the SphereFace2 loss with the same hyperparameters as in the main paper ($\lambda = 0.7$, $r = 40$, $m = 0.4$ for the CosFace-style additive margin, and $t = 3.0$). For our Unified KD experiments, we set the ILED parameters to $r = 40$, $s = 0.9$, $\lambda_{\mathrm{ILED}} = 9$, and the RPSD parameters to $r' = 60.0$, $t = 0.05$, $\lambda_{\mathrm{RPSD}} = 40.0$, using a memory-bank size of $q = 3$. All results were reported using model weights saved at 140k training iterations with a batch size of 64.

Table 4. Teacher–student distillation matrix. The baseline without knowledge distillation is denoted by (✗), and Unified KD (ILED + RPSD) is denoted by (✓). All numbers represent accuracies (%) on the LFW, AgeDB, CA-LFW, and CP-LFW datasets.

Teacher	Student	Scheme	LFW	AgeDB	CA-LFW	CP-LFW
ResNet100	–	Teacher Only	99.633	95.550	94.500	92.950
	ResNet18	✗	99.417	93.367	93.417	90.917
		✓	99.600	94.700	93.900	91.667
	MobileFaceNet	✗	98.917	88.683	91.433	87.300
		✓	99.417	92.650	93.217	90.117
	ResNet34	✗	99.617	95.017	94.217	92.333
		✓	99.650	95.533	94.283	93.183
	ResNet50	✗	99.583	95.233	94.300	92.600
		✓	99.633	95.850	94.417	93.550
ResNet50	–	Teacher Only	99.583	95.233	94.300	92.733
	ResNet18	✗	99.417	93.367	93.417	90.917
		✓	99.550	94.600	93.933	91.767
	MobileFaceNet	✗	98.917	88.683	91.433	87.300
		✓	99.417	92.717	93.100	90.100
	ResNet34	✗	99.617	95.017	94.217	92.333
		✓	99.667	95.417	94.267	92.883
DPN98	–	Teacher Only	99.233	90.617	92.550	90.083
	ResNet18	✗	99.417	93.367	93.417	90.917
		✓	99.500	92.500	93.267	90.800
	MobileFaceNet	✗	98.917	88.683	91.433	87.300
		✓	99.383	90.750	92.417	90.083
	ResNet34	✗	99.617	95.017	94.217	92.333
		✓	99.667	93.717	93.700	92.550
	ResNet50	✗	99.583	95.233	94.300	92.600
		✓	99.650	93.917	93.817	92.800

When the teacher is strong (e.g. ResNet100) compared to the student, Unified KD delivers substantial improvements over without KD method, demonstrating that ILED + RPSD effectively transfers richer teacher embeddings into a smaller model. Interestingly, in some pairings, the Unified KD student even surpasses the teacher's native performance, suggesting that the distillation losses also regularize and smooth teacher overconfidence. Conversely, when the teacher is weaker than the student (for example, DPN98 teaching ResNet18), applying Unified KD sometimes yields worse results than training without distillation. This indicates that when the teacher's embeddings are less discriminative than the student's capacity, the additional distillation losses can overconstrain the student and lead to suboptimal fitting.

C.2. Effect of Teacher Loss Function on Student Performance

In Table 5, the teacher model consistently uses the ResNet100 architecture with pretrained weights from an open-source repository, each trained with different FR losses. The student model uses ResNet18 trained with SphereFace2 as the FR loss. The proposed KD method maintains consistency across pretrained models and losses. Note that the lower accuracies for the SphereFace2-pretrained teacher are because it was trained for significantly fewer iterations than the other open-source teacher models.

Table 5. Performance of the student model trained with pretrained teacher models using various face recognition (FR) loss functions, evaluated on multiple validation datasets.

FR Loss	**LFW**	**AgeDB**	**CA-LFW**	**CP-LFW**
ArcFace	99.517	94.350	93.850	91.383
AdaFace	99.500	95.567	94.500	91.733
SphereFace2	99.417	93.367	93.417	90.917

C.3. Baseline Implementation Details

For the L2 raw features and FC loss, we set the loss weights to $\lambda_{L2} = 0.1$ (to compensate for the large L2 values) and $\lambda_{FC} = 3$. In Standard Temperature-Scaled KD, we used a temperature of $T = 3$. For GKD, we chose $\tau = 0.93$, $\lambda_1 = 8$, $\lambda_2 = 1$, and $T = 1$. For ShrinkTeaNet Angular KD, we assigned the final-layer weight $\lambda_n = 1$ and recursively defined each intermediate weight by $\lambda_i = \frac{1}{2}\,\lambda_{i+1}$ for $i = n-1, \ldots, 1$. DarkRank employs a scale factor $\alpha = 3$, exponent $\beta = 3$, and a list-wise KL loss weight of $\lambda = 0.1$. For EC-KD, we used λ_1=0.0005 and λ_2= $2 \times \lambda_1$, and due to the very small magnitude of the FC loss (in the range of 10^{-10}), we normalized the features before computing the loss. Finally, in Correlation Congruence KD (CCKD), we used $\gamma = 0.5$, $\beta = 10$, $\alpha = 0.5$, and replaced KL divergence with an L2-mimic loss as recommended in the paper. Additionally, we fine-tuned the hyperparameters of these methods to further enhance their performance.